```
EEEEEEEEEEEEEEEE  DDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD     TTTTTTTTTTTTTTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEEEEEEEEEEE      DDD    DDD           TTT
EEEEEEEEEEEE      DDD    DDD           TTT
EEEEEEEEEEEE      DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEE               DDD    DDD           TTT
EEEEEEEEEEEEEEEE  DDDDDDDDDD           TTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD          TTT
EEEEEEEEEEEEEEEEE  DDDDDDDDDD          TTT
```

```
  SSSSSSSS    CCCCCCC    RRRRRRR    NN      NN    CCCCCCC    000000    LL
  SSSSSSSS    CCCCCCC    RRRRRRR    NN      NN    CCCCCCC    000000    LL
SS            CC         RR    RR   NN      NN   CC          00    00  LL
SS            CC         RR    RR   NN      NN   CC          00    00  LL
SS            CC         RR    RR   NNNN    NN   CC          00    00  LL
SS            CC         RR    RR   NNNN    NN   CC          00    00  LL
  SSSSSS      CC         RRRRRRRR   NN  NN  NN   CC          00    00  LL
  SSSSS       CC         RRRRRRRR   NN  NN  NN   CC          00    00  LL
        SS    CC         RR  RR     NN    NNNN   CC          00    00  LL
        SS    CC         RR  RR     NN    NNNN   CC          00    00  LL
        SS    CC         RR    RR   NN      NN   CC          00    00  LL          ::::
        SS    CC         RR    RR   NN      NN   CC          00    00  LL          ::::
SSSSSSSS      CCCCCCCC   RR    RR   NN      NN    CCCCCCC    000000    LLLLLLLLLL   ::::
SSSSSSSS      CCCCCCCC   RR    RR   NN      NN    CCCCCCC    000000    LLLLLLLLLL   ::::


LL             IIIIII     SSSSSSSS
LL             IIIIII     SSSSSSSS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II       SSSSSS
LL               II       SSSSSS
LL               II             SS
LL               II             SS
LL               II             SS
LL               II             SS
LLLLLLLLLL     IIIIII     SSSSSSS
LLLLLLLLLL     IIIIII     SSSSSSSS
```

```
    1    0001  0  %TITLE 'EDT$SCRNCOL - compute new column position'
    2    0002  0  MODULE EDT$SCRNCOL (                          ! Compute new column position
    3    0003  0             IDENT = 'V04-000'                  ! File: SCRNCOL.BLI Edit: JBS1003
    4    0004  0             ) =
    5    0005  1  BEGIN
    6    0006  1
    7    0007  1  !****************************************************************************
    8    0008  1  !*                                                                          *
    9    0009  1  !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
   10    0010  1  !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
   11    0011  1  !*  ALL RIGHTS RESERVED.                                                    *
   12    0012  1  !*                                                                          *
   13    0013  1  !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
   14    0014  1  !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
   15    0015  1  !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
   16    0016  1  !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
   17    0017  1  !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
   18    0018  1  !*  TRANSFERRED.                                                            *
   19    0019  1  !*                                                                          *
   20    0020  1  !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
   21    0021  1  !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
   22    0022  1  !*  CORPORATION.                                                            *
   23    0023  1  !*                                                                          *
   24    0024  1  !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
   25    0025  1  !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
   26    0026  1  !*                                                                          *
   27    0027  1  !*                                                                          *
   28    0028  1  !****************************************************************************
   29    0029  1
   30    0030  1
   31    0031  1  !++
   32    0032  1  ! FACILITY:     EDT -- The DEC Standard Editor
   33    0033  1  !
   34    0034  1  ! ABSTRACT:
   35    0035  1  !
   36    0036  1  !      This module computes a new column position for vertical moves.
   37    0037  1  !
   38    0038  1  ! ENVIRONMENT:  Runs at any access mode - AST reentrant
   39    0039  1  !
   40    0040  1  ! AUTHOR: Bob Kushlis, CREATION DATE: September 8, 1979
   41    0041  1  !
   42    0042  1  ! MODIFIED BY:
   43    0043  1  !
   44    0044  1  ! 1-001 - Original. DJS 12-Feb-1981.  This module was created by
   45    0045  1  !         extracting the routine EDT$$SC CPUNEWCOLPOS  from module SCREEN.
   46    0046  1  ! 1-002 - Regularize headers.  JBS 13-Mar-1981
   47    0047  1  ! 1-003 - Avoid using the screen data base, since the current line may not
   48    0048  1  !         be on the screen.  JBS 10-Sep-1982
   49    0049  1  !--
   50    0050  1
```

```
   52    0051  1   %SBTTL 'Declarations'
   53    0052  1   !
   54    0053  1   ! TABLE OF CONTENTS:
   55    0054  1   !
   56    0055  1
   57    0056  1   REQUIRE 'EDTSRC:TRAROUNAM';
   58    0495  1
   59    0496  1   FORWARD ROUTINE
   60    0497  1       EDT$$SC_CPUNEWCOLPOS : NOVALUE;
   61    0498  1
   62    0499  1   !
   63    0500  1   ! INCLUDE FILES:
   64    0501  1   !
   65    0502  1
   66    0503  1   REQUIRE 'EDTSRC:EDTREQ';
   67    0638  1
   68    0639  1   !
   69    0640  1   ! MACROS:
   70    0641  1   !
   71    0642  1   !       NONE
   72    0643  1   !
   73    0644  1   ! EQUATED SYMBOLS:
   74    0645  1   !
   75    0646  1   !       NONE
   76    0647  1   !
   77    0648  1   ! OWN STORAGE:
   78    0649  1   !
   79    0650  1   !       NONE
   80    0651  1   !
   81    0652  1   ! EXTERNAL REFERENCES:
   82    0653  1   !
   83    0654  1   !       In the routine
```

```
 85    0655  1  %SBTTL 'EDT$$SC_CPUNEWCOLPOS  - compute new column position'
 86    0656  1
 87    0657  1  GLOBAL ROUTINE EDT$$SC_CPUNEWCOLPOS                          ! Compute new column position
 88    0658  1     : NOVALUE =
 89    0659  1
 90    0660  1  !++
 91    0661  1  ! FUNCTIONAL DESCRIPTION:
 92    0662  1  !
 93    0663  1  !     This routine computes a new column position for vertical moves.  When
 94    0664  1  !     any command other than a vertical arrow which moves the cursor is executed
 95    0665  1  !     this procedure is called to get the new column position for vertical moves.
 96    0666  1  !     The column position is kept in EDT$$G_COL .
 97    0667  1  !
 98    0668  1  ! FORMAL PARAMETERS:
 99    0669  1  !
100    0670  1  !     NONE
101    0671  1  !
102    0672  1  ! IMPLICIT INPUTS:
103    0673  1  !
104    0674  1  !     EDT$$T_LN_BUF
105    0675  1  !     EDT$$A_LN_PTR
106    0676  1  !
107    0677  1  ! IMPLICIT OUTPUTS:
108    0678  1  !
109    0679  1  !     EDT$$G_COL
110    0680  1  !
111    0681  1  ! ROUTINE VALUE:
112    0682  1  !
113    0683  1  !     NONE
114    0684  1  !
115    0685  1  ! SIDE EFFECTS:
116    0686  1  !
117    0687  1  !     NONE
118    0688  1  !
119    0689  1  !--
120    0690  1
121    0691  2     BEGIN
122    0692  2
123    0693  2     EXTERNAL ROUTINE
124    0694  2         EDT$$FMT_CHWID;                                    ! Compute the width of a character
125    0695  2
126    0696  2     EXTERNAL
127    0697  2         EDT$$G_COL,                                        ! The column we should be on.
128    0698  2         EDT$$T_LN_BUF,                                     ! Current line buffer.
129    0699  2         EDT$$A_LN_PTR;                                     ! Current character pointer.
130    0700  2
131    0701  2     LOCAL
132    0702  2         COL_POINT;
133    0703  2
134    0704  2     COL_POINT = EDT$$T_LN_BUF;
135    0705  2     EDT$$G_COL = 0;
136    0706  2
137    0707  2     WHILE CH$PTR_NEQ (.COL_POINT, .EDT$$A_LN_PTR) DO
138    0708  2         EDT$$G_COL = .EDT$$G_COL + EDT$$FMT_CHWID (CH$RCHAR_A (COL_POINT), .EDT$$G_COL);
139    0709  2
140    0710  1     END;                                                  ! of routine EDT$$SC_CPUNEWCOLPOS
```

```
;                                                            .TITLE   EDT$SCRNCOL EDT$SCRNCOL - compute new column po
                                                                               sition
                                                             .IDENT   \V04-000\

                                                             .EXTRN   EDT$$FMT_CHWID, EDT$$G_COL
                                                             .EXTRN   EDT$$T_LN_BUF, EDT$$A_LN_PTR

                                                             .PSECT   _EDT$CODE,NOWRT,  SHR,  PIC,2

                                   000C 00000                .ENTRY   EDT$$SC_CPUNEWCOLPOS, Save R2,R3              ; 0657
                  53 00000000G  00  9E 00002                 MOVAB    EDT$$G_COL, R3
                  52 00000000G  00  9E 00009                 MOVAB    EDT$$T_LN_BUF, COL_POINT                     ; 0704
                            63  D4 00010                      CLRL     EDT$$G_COL                                  ; 0705
           00000000G  00    52  D1 00012 1$:                 CMPL     COL_POINT, EDT$$A_LN_PTR                    ; 0707
                            11  13 00019                      BEQL     2$
                            63  DD 0001B                      PUSHL    EDT$$G_COL                                  ; 0708
                        7E  82  9A 0001D                      MOVZBL   (COL_POINT)+, -(SP)
           00000000G  00  02  FB 00020                        CALLS    #2, EDT$$FMT_CHWID
                        63  50  C0 00027                      ADDL2    R0, EDT$$G_COL
                            E6  11 0002A                      BRB      1$
                            04 0002C 2$:                      RET                                                 ; 0710
```

; Routine Size: 45 bytes,   Routine Base: _EDT$CODE + 0000

```
;  141          0711  1
;  142          0712  1 !<BLF/PAGE>
```

```
; 144          0713 1 END                                          ! of module EDT$SCRNCOL
; 145          0714 1
; 146          0715 0 ELUDOM
```

```
                              PSECT SUMMARY

        Name                      Bytes                      Attributes

  _EDT$CODE                         45   NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
```

```
                        Library Statistics

                                      -------- Symbols --------      Pages         Processing
        File                          Total   Loaded   Percent      Mapped        Time

  _$255$DUA28:[EDT.SRC]EDT.L32;1        377       1        0          40           00:00.2
  _$255$DUA28:[EDT.SRC]PSECTS.L32;1       2       1       50           7           00:00.1
```

```
                              COMMAND QUALIFIERS

      BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:SCRNCOL/OBJ=OBJ$:SCRNCOL MSRC$:SCRNCOL.BLI/UPDATE=(ENH$:SCRNCOL
      )
```

```
; Size:             45 code + 0 data bytes
; Run Time:           00:09.4
; Elapsed Time:       00:11.7
; Lines/CPU Min:      4544
; Lexemes/CPU-Min: 12165
; Memory Used:  56 pages
; Compilation Complete
```

SCRRESET
LIS

SCRRLIN
LIS

SCRRELPOS
LIS

SCRINSERT
LIS

SCRNOSCR
LIS

SCRNCOL
LIS

SCRMCOL
LIS

SCRUPDATE
LIS

SCRINIT
LIS

SCRNEWINS
LIS

SCRREGION
LIS

SCRREPMOD
LIS

SCRREV
LIS

SCRNEWDEL
LIS

SCRNOREV
LIS

SCRMOVETO
LIS

SCRINSMOD
LIS